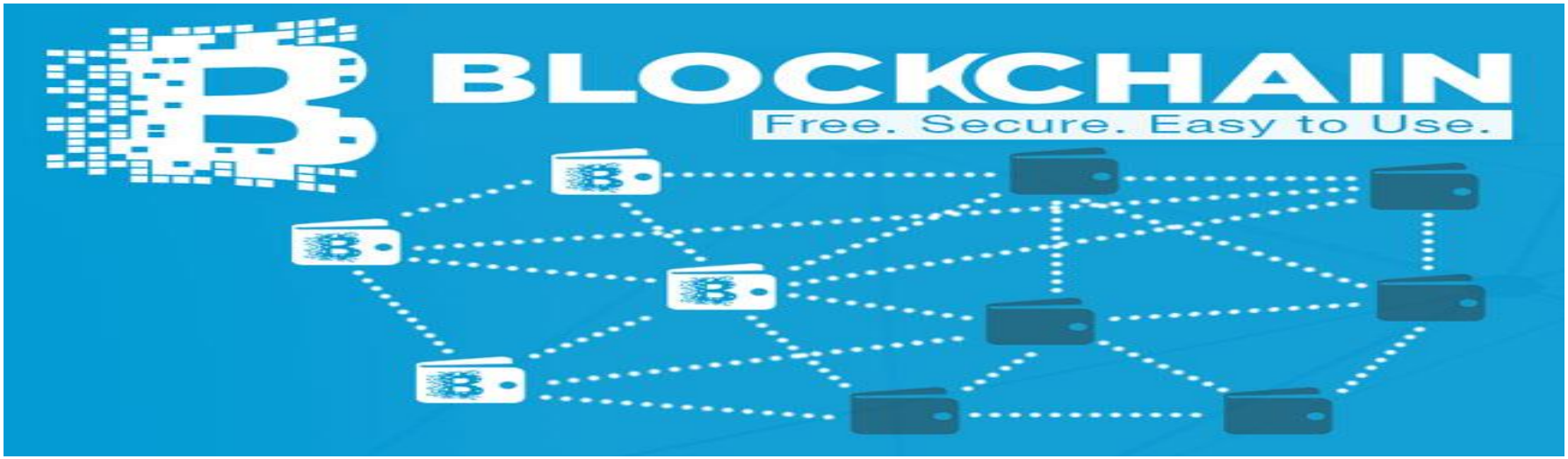
SCIM

Beyond the hype: Block- chain in the Supply-chain

Ian Robertson
Department of Computer Science, Warwick University

mailto:I.robertson@warwick.ac.uk



Hype and reality

Royal Mint to issue c

29 November 2016 | 8929 views | 1

Top of the News

Bloomberg Report Describes Supply Chain Nightmare
(October 4, 2018)

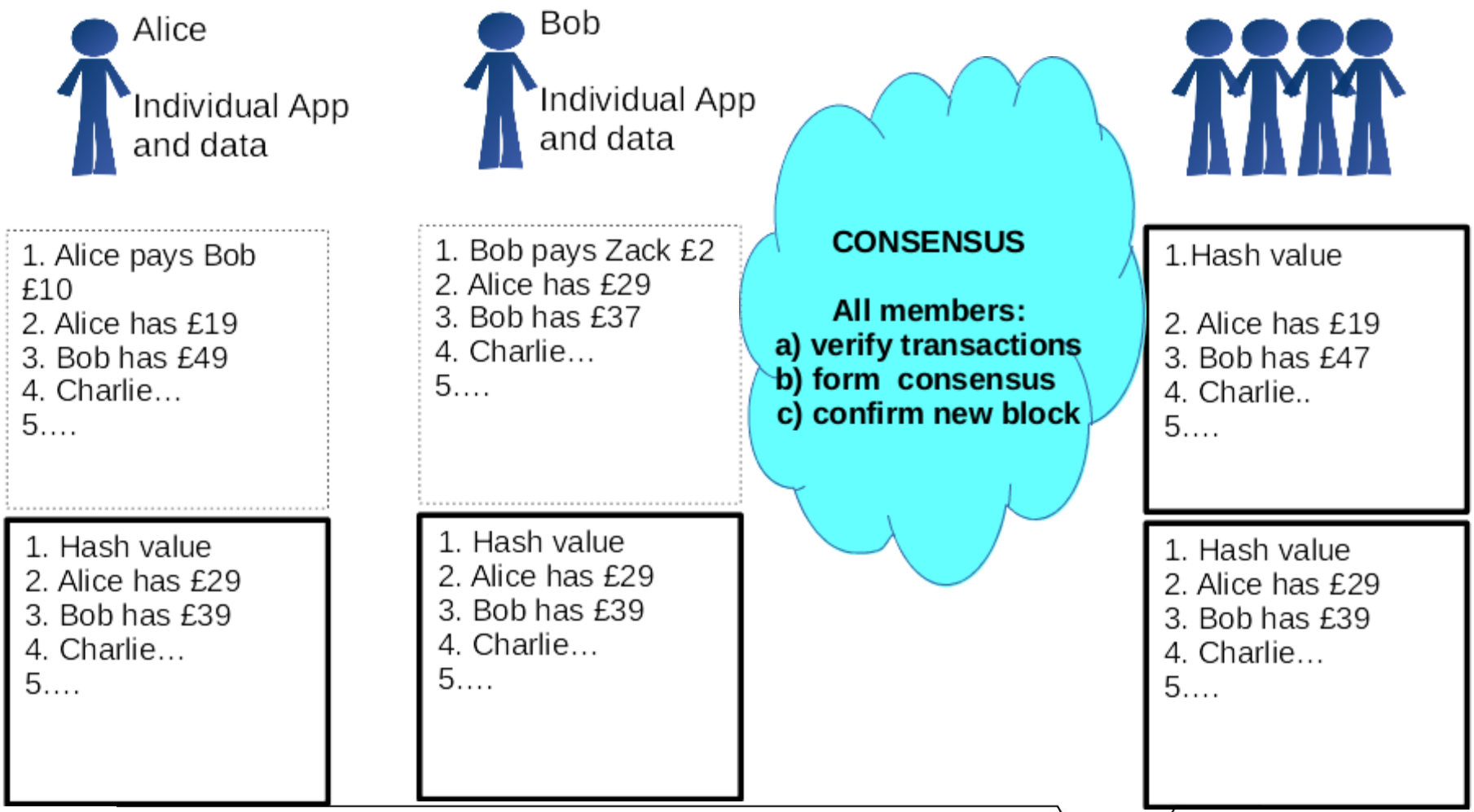
Industry-wide blockchain breakthroughs at least six years away

Facebook gets big backers for Glo project

While three quarters of banks are actively experimenting with blockchain technology, regulatory uncertainty and a lack of inhouse expertise are proving major roadblocks to near-term adoption, accordin...

WARWICK
THE UNIVERSITY OF WARWICK

Block-chains – data structures to support distributed applications



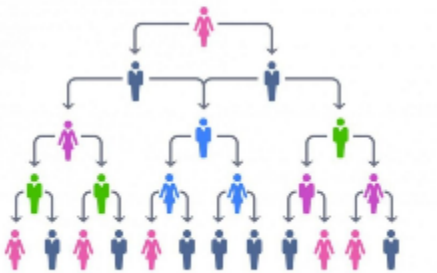
Ways to get a consensus

Appoint a "leader"

Central authority ("permissioned")

- Fast
- Accurate
- Reliable
- Requires trust
- Subject to legal control

Hybrid or hierarchic



Bottom up pairwise

- Potentially very fast

No formal proof yet

Run a "competition"

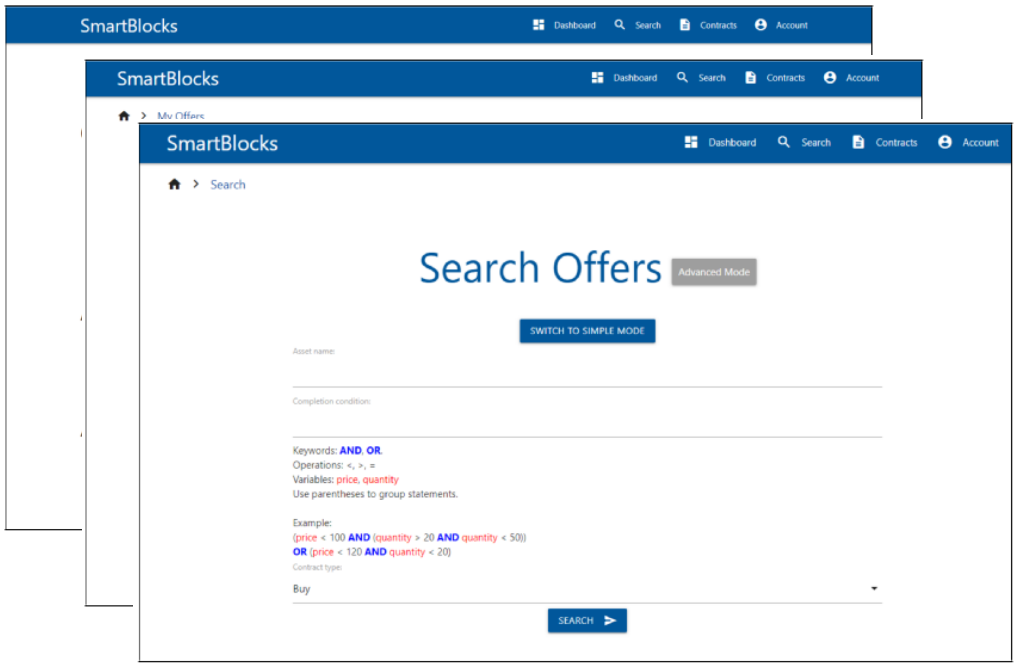
- Slow and Inefficient
- Security flaws
- No central authority !

E.g. Bitcoin

1) "Miners" race to solve a computer problem.
2) Winner adds the new block

- This is the key issue for a distributed, asynchronous system
- Systems that are reliable tend to be slow (1 TPS on average)
- Distribution introduces new security flaws

“Smart-contracts” - automatic matching and execution of contracts



Keywords: **AND**, **OR**.
Operations: <, >, =
Variables: price, quantity
Use parentheses to group statements.

Example:
(price < 100 **AND** (quantity > 20 **AND** quantity < 50))
OR (price < 120 **AND** quantity < 20)

Price < 100

Price < 100 AND Quantity < 40

Expr AND ((Expr) OR Expr)

[pq][<=>][\d] + ([.][\d]+)?

Outline process:

1) Create offers to sell
2) Create requests to buy

3) Consensus shares offers

4) Match offers / requests
5) Propose “deals”

6) Consensus finalizes contracts

What works and what doesn't ?

Smart-contracts

- Smart-languages
- Low interoperability
- Language restrictions
- Cheap(ish)

Etherium, HyperLedger Ripple or hybrid

1) Events and triggers
2) Trade matching
3) Contract signing and encryption

Open, Distributed systems

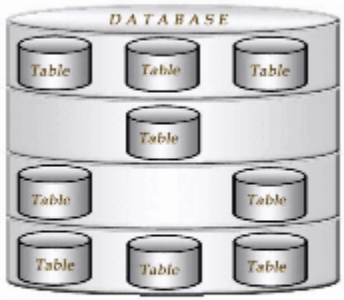
Uses distributed processing to increase speed and security

Uses consensus process to support optimisation

Runs multiple smart-contract languages.

Smarter

Traditional



- High speed (1000 TPS)
- High security - understood
- Expensive, centralized

IBM / Oracle / MySQL

1) Data management
2) Events and triggers
3) Signing and encryption

Open, distributed ("un-permissioned") Blockchain

Bitcoin, XXXcoin

1) Crypto-currency transactions
2) Exchange of value

- Slow (1TPS) and Long latency (5-10 minutes)
- Security and reliability risks
- Pseudo-anonymous
- No central control (no legal authority)

More distributed

WARWICK
THE UNIVERSITY OF WARWICK

Central or distributed?

Permissioned or open?

How does the consensus work?

How fast is it?

How secure is it?